

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Фізико-математичний факультет
(повна назва інституту/факультету)

Кафедра загальної фізики та фізики твердого тіла
(повна назва кафедри)

«На правах рукопису»
УДК 538.911:004](043.3)

«До захисту допущено»

Завідувач кафедри
_____ В.Й. Котовський
(підпис) (ініціали, прізвище)

“ ____ ” _____ 20__ р.

Магістерська дисертація

зі спеціальності (спеціалізації) 104 фізика та астрономія _____
(код і назва спеціальності)

на тему: Моделювання динаміки розпаду вістря вольфрамових голок _____

Виконав (-ла): студент (-ка) 2 курсу магістерського рівня, групи ОФ-71мп
(шифр групи)

Заводовський Михайло Юрійович _____
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник д. фіз. мат. н., проф. Горшков В. М. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант _____
(назва розділу) (науковий ступінь, вчене звання, , прізвище, ініціали) (підпис)

Рецензент д. фіз. мат. н., проф. Решетняк С. О. _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць інших
авторів без відповідних посилань.

Студент _____
(підпис)

ЗАВДАННЯ

на магістерську роботу студента

Заводовського Михайла Юрійовича

1. Тема роботи: Моделювання динаміки розпаду вістря вольфрамівих голок

Керівник роботи: д. фіз. мат. н., проф. Горшков В. М.

Затверджено наказом по університету від «02» листопада 2018р. №4064-с

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи: використання розробленого програмного забезпечення дозволяє визначити стійкість структур з ОЦК ґраткою до розпаду.

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити):

- Ознайомитися з процесом поверхневої дифузії та будовою ОЦК ґратки.
- Ознайомитися з чисельною моделю руху вільних та зв'язаних атомів.
- Розробити комп'ютерну програму для розрахунку динаміки розпаду вістря вольфрамівих голок.
- Порахувати тестову задачу, для перевірки чисельної моделі
- Порахувати задачу розпаду вольфрамової голки для декількох випадків.
- Побудувати графіки, які описують процеси розпаду.

5. Перелік графічного матеріалу: 14 малюнків

6. Дата видачі завдання 03.09.2018р.

Календарний план

Назва етапів виконання магістерської дисертації	Строк виконання етапів роботи	Примітка
Перегляд літературних джерел, складання огляду літератури	03.09.2018-09.09.2018	
Пошук і добір фактичних матеріалів, їх групування та систематизація	10.09.2018-23.09.2018	
Написання коду комп'ютерної програми для розрахунку розпаду вістря вольфрамівих голок	10.09.2018-26.10.2018	
Запуск показового експерименту задля перевірки працездатності зробленої математичної моделі	27.10.2018-31.10.2018	
Виконання математичних обчислень згідно зробленої раніше математичної моделі	01.11.2018-07.11.2018	
Обробка та візуалізація отриманих результатів	08.11.2018-18.11.2018	
Опис отриманих результатів	19.11.2018-25.11.2018	
Попередній захист	12.12.2018	
Корегування роботи відповідно до зауважень комісії	12.12.2018-17.12.2018	
Остаточний захист роботи	18.12.2018	

Магістр _____
(підпис)

Заводовський М. Ю.
(ініціали, прізвище)

Керівник роботи _____
(підпис)

Горшков В. М.
(ініціали, прізвище)

Реферат

Об'єкт дослідження: вольфрамові голки нанорозмірів.

Мета роботи: розробка програмного забезпечення для розрахунку динаміки розпаду вольфрамових голок, отримання результатів розрахунку.

Методи дослідження:

теоретичні: узагальнення даних стосовно теми дослідження на основі науково-методичної літератури та офіційних освітньо-наукових джерел;

практичні: написання коду програми для реалізації процесів розпаду вольфрамових голок.

Завдання дослідження:

- Ознайомитися з процесом поверхневої дифузії та будовою ОЦК гратки.
- Ознайомитися з чисельною моделю руху вільних та зв'язаних атомів.
- Розробити комп'ютерну програму для розрахунку динаміки розпаду вістря вольфрамових голок.
- Порахувати тестову задачу, щоб упевнитися в правильності роботи програми.
- Порахувати задачу для декількох випадків.
- Побудувати графіки отриманих результатів.

Результати та їх новизна

Розроблена комп'ютерна програма для розрахунку динаміки розпаду вістря вольфрамових голок.

Отримана залежність розпаду голки на краплі від її довжини(кута вістря).

Ступінь впровадження: робота є науковою базою для подальших досліджень.

Рекомендації щодо використання результатів роботи та область їх застосування: результати роботи можуть бути використані в мікро- та наноелектроніці, для пристроїв де використовуються голки нанорозмірів.

Значення роботи та висновки: розроблена комп'ютерна модель розпаду вістря вольфрамових голок для різних параметрів.

Ключові слова: *Поверхнева дифузія, ОЦК ґратка, чисельна модель, графік.*

В роботі наведено: використаної літератури - 29, сторінок: - 67, рисунків-14.

Abstract

Object of research: tungsten needles of nanoscale

Purpose: development of software for calculating the dynamics of decay of tungsten needles, obtaining the results of calculation.

Research methods:

theoretical: generalization of data on the topic of research on the basis of scientific and methodological literature and official educational-scientific sources;

Practical: writing a program code for the implementation of the processes of decomposition of tungsten needles.

Objectives of the study:

- To familiarize with the process of surface diffusion and structure of bcc lattice.
- Look at the numerical model of the motion of free and bound atoms.
- Develop a computer program to calculate the dynamics of the decay of the tip of the tungsten needles.
- Calculate the test task to ensure that the programs work correctly.
- Calculate the task for several occasions.
- Build graphs of the results.

Results and their novelty

A computer program for calculating the dynamics of the decay of the tip of the tungsten needles has been developed.

Obtained dependence of the disintegration of the needle on the drop from its length .

Degree of implementation: work is a scientific basis for further research.

Recommendations on the use of the results of work and the scope of their application: the results of work can be used in micro and nanoelectronics, for devices where nano-sized needles are used.

Value of work and conclusions: a computer model of the collapse of the tip of tungsten needles for various parameters was developed.

Keywords: surface diffusion, bcc lattice, numerical model, graphic.

The paper contains: used literature -29, pages: -67, drawings -14.

ЗМІСТ

ВСТУП	9
РОЗДІЛ I. Дифузія.....	10
1.1. Самодифузія.....	10
1.2. Поверхнева дифузія.....	13
1.3. Коефіцієнт дифузії невзаємодіючих частинок	14
1.4. Коефіцієнт дифузії взаємодіючих частинок: взаємодія найближчих сусідів	15
РОЗДІЛ II. Кристалічні ґратки металів.....	18
2.1. Основні типи кристалічних решіток металів.....	18
2.2. Кристалографічні напрямки і площини	22
РОЗДІЛ III. Моделювання динаміки розпаду вістря вольфрамових голок ...	24
3.1. Опис чисельної моделі	24
3.2. Розпад вістря вольфрамових голок на краплі	33
ВИСНОВКИ.....	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	41
ДОДАТКИ.....	44

ВСТУП

Дифузія – це процес руху молекул, що за нормальних умов ніколи не припиняється. Завдяки певним аспектам цього процесу перехід речовини із рідкого/газоподібного стану переходить у твердий із певними особливостями, наприклад, утворюючи певні структури. Також і там дифузія не припиняється, і деякі структури виявляються стійкими, а деякі ні. Які саме – визначається параметрами системи і структури.

Процеси поверхневої дифузії у твердих тілах є предметом досліджень у галузі фізики твердого тіла. Поверхнева дифузія відіграє визначальну роль у процесах росту тонких плівок, формуванні наноструктур на поверхні підкладок і спікання кераміки. Процеси поверхневої дифузії у твердих тілах досліджуються досить широко. Однак процеси розпаду наноструктур, таких, як наноголки та нанострижні залежить від багатьох процесів, і не є достатньо дослідженою проблемою. Тому *актуальним* є подальше дослідження цих процесів.

Складність процесів, що відбуваються при розпаді наноголки на краплі, не дає можливість одержати їхній строгий аналітичний опис. Розвиток комп'ютерної техніки протягом останніх десятиліть дає можливість використовувати її для вивчення таких процесів за допомогою моделювання динаміки цих процесів.

РОЗДІЛ I. Дифузія

1.1. Самодифузія

Самодифузія – це явище перенесення в однорідному газі з односторонніх молекул за наявності градієнта концентрації n . Під час самодифузії величиною A , яка зазнає перенесення, є концентрація n , а саме $A = n/n_0$, де n_0 – рівноважна концентрація молекул у газі.

Тоді давайте візьмемо і частину їх помітимо. Пофарбувати їх у різні кольори не вдасться, а от використати радіоактивний ізотоп цього ж газу цілком можливо. Така помітка ніяким чином не вплине ні на взаємодію молекул, ні на їхній рух [1]. Концентрацію мічених молекул позначимо як n_1 , немічених – n_2 . І, поки ми не зняли перегородку, $n_1 = n_2 = n$.

Знімаємо перегородку. Молекули внаслідок теплового руху будуть перемішуватись. Величина концентрації змінюється вздовж вибраного напрямку $n_1 = n_1(x)$. Для концентрацій мічених і не мічених молекул виникає співвідношення

$$n_1(x) + n_2(x) = n;$$

$$-\frac{dn_1}{dx} = \frac{dn_2}{dx}.$$

Виділимо в об'ємі площадку площею dS , перпендикулярну осі x .

Як і у випадку теплопровідності, вираз для маси, перенесеної через площадку, можна записати із загальних міркувань, але цей закон підтверджений експериментально і має назву *закону Фіка*. Він полягає у тому, що перенесена за одиницю часу маса речовини пропорційна масі молекули речовини, градієнту її концентрації та площі площадки, через яку речовина переноситься :

$$dM_1 = -Dm \frac{dn_1}{dx} dS = -D \frac{d\rho_1}{dx} dS.$$

Або можна ще сказати, що вона пропорційна градієнту густини речовини та площі площадки.

В нашому випадку dM_1 – маса мічених молекул, що переноситься через площадку. Вона визначається як mdN_1 , тоді дифузійний потік молекул через площадку

$$dN_1 = \frac{dM_1}{m} = -D \frac{dn_1}{dx} dS.$$

Коефіцієнтом пропорційності є величина D , що має назву *коефіцієнта дифузії*. У такому випадку *коефіцієнт дифузії визначається дифузійним потоком, при одиничному градієнті концентрації*.

Знову звернемось до основної формули переносу

$$dG = -\frac{n\bar{v}}{3} \lambda \frac{dg}{dx} dS.$$

Ми вже бачили, що g характеризує величину, яку переносить 1 молекула. У випадку в'язкості це був імпульс $g = mu$, у випадку теплопровідності це середня кінетична енергія поступального руху молекули $g = \frac{3}{2}kT$. Маса, яку можуть перенести через площадку мічені молекули, становить mn_1 , тоді на одну молекулу газу припадає маса $m \frac{n_1}{n}$. Якщо ми візьмемо рівняння Фіка не для потоку маси, а для дифузійного потоку, то у якості величини, що переноситься, буде виступати відносна концентрація мічених молекул

$$g(x) = \frac{n_1}{n}.$$

Тоді основна формула набуває вигляду

$$dG = -\frac{\bar{n}\bar{v}}{3}\lambda \frac{1}{n} \frac{dn_1}{dx} dS = -\frac{\bar{v}\lambda}{3} \frac{dn_1}{dx} dS = dN_1.$$

Порівнявши із формулою Фіка

$$dN_1 = -D \frac{dn_1}{dx} dS,$$

отримаємо вираз для коефіцієнта самодифузії дифузії

$$D = \frac{\bar{v}\lambda}{3}$$

Знову ж таки аналізуємо залежності

$$1. D = f(p).$$

$$2. D = f(T).$$

Швидкість від тиску не залежить, довжина вільного пробігу $\lambda \sim \frac{1}{p}$, отже

і $D \sim \frac{1}{p}$. І, звичайно, формули не працюють при $\lambda \gg d$. Тут не можна

говорити про дифузію, оскільки перенос маси відбувається зі швидкістю руху молекул, а отже, $D \neq f(p)$.

А температурна залежність, як і для коефіцієнтів в'язкості і теплопровідності, визначається залежністю швидкості від температури

$\bar{v} \sim \sqrt{T}$ та довжини вільного пробігу від температури. Отже, $D \sim \frac{\sqrt{T}}{1 + \frac{S}{T}}$.

1.2. Поверхнева дифузія

Наявність чужорідних атомів (далі адатомів) на поверхні твердого тіла істотно змінює його характеристики [2]. Швидкість переміщення адатомів по поверхні істотно залежить від глибини потенціальних мінімумів, утворених атомами підкладки. Характер цього процесу визначається співвідношенням енергій адатом-адатомної (а-а) взаємодії та взаємодії адатомів з підкладкою (а-п). Якщо енергія а-а взаємодії (φ) істотно перевищує висоту потенціального рельєфу підкладки (E_a), то у випадку відштовхувальної а-а взаємодії адатоми намагатимуться відійти один від одного на максимальну відстань, утворюючи на поверхні гексагональну структуру. Далі розглянемо протилежну ситуацію, показану на рис. 1, при якій $E_a > \varphi$. Тут наявність близьких сусідів збільшує ймовірність подолання активаційного бар'єру. З метою математичної формалізації процесу міграції вважатимемо, що $E_a \gg T$.

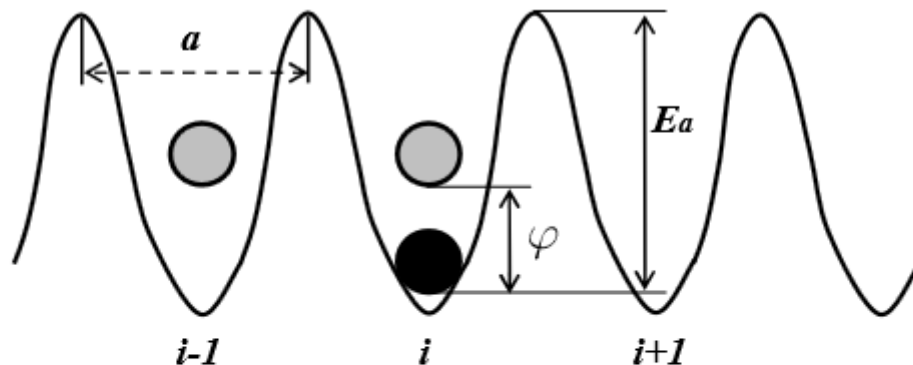


Рис. 1. Одновимірний потенціальний рельєф. Для переходу одинокій частинці (чорний кружок в i -му мінімумі) у вільний $i+1$ -ий мінімум необхідно подолати активаційний бар'єр E_a . Якщо ж у $i-1$ -му мінімумі знаходиться частинка (сірий кружок) і існує відштовхувальна а-а взаємодія φ , то активаційний бар'єр зменшується до величини $E_a - \varphi$.

Тоді частинки перебуватимуть протягом довгого часу в мінімумах потенціальної енергії (пронумерованих на рис. 1 числами $\dots i-1, i, i+1 \dots$) і лише інколи “перестрибуватимуть” у сусідні незаповнені мінімуми. Стан такої системи задається числами заповнення n_i , де кожне з чисел n_i дорівнює 1 чи 0

в залежності від того чи є частинка в i -му мінімумі ($n_i = 1$), чи відсутня ($n_i = 0$).

Зміна з часом величин n_i визначається рівнянням балансу:

$$n_i(t + \Delta t) - n_i(t) = -\Delta t \sum_j [\nu_{ij} n_i (1 - n_j) - \nu_{ji} n_j (1 - n_i)], \quad (1)$$

де ν_{ij} ймовірність переходу частинки з вузла i у вузол j за одиницю часу; інтервал часу Δt беруть значно більшим, ніж тривалість стрибка t , проте значно меншим, ніж час перебування у вузлі $(\nu_{ij})^{-1}$. Виконання критерію $(\nu_{ij})^{-1} \gg t$, який можна назвати наближенням миттєвих стрибків, є необхідною умовою коректного опису міграції адатомів на основі моделі решіткового газу. Далі вважатимемо, що стрибки відбуваються лише між найближчими вузлами.

1.3. Коефіцієнт дифузії невзаємодіючих частинок

Коли взаємодія між частинками відсутня, то $\nu_{ij} = \nu_{ji} = \nu$. Тоді, одержимо:

$$\frac{n_i(t + \Delta t) - n_i(t)}{\Delta t} = -\sum_j \nu (n_i - n_j). \quad (2)$$

Нехай густина адатомів на поверхні є пивною функцією, що змінюється на масштабах, значно більших за a . Тоді можна усереднити (2) по інтервалу часу t_D , який значно більший за $(\nu_{ij})^{-1}$, але протягом якого середнє заповнення вузлів майже не зміниться. Таке усереднення позначимо тильдою. Після цього рівність (2) можна записати у вигляді

$$\frac{\partial \widetilde{n_i(t)}}{\partial t} = -\sum_j \nu (\widetilde{n_i} - \widetilde{n_j}), \quad (3)$$

де член у лівій частині вже можна замінити на похідну по часу [3].

Далі розглянемо найпростіший випадок квадратної решітки. Переходячи до звичніших позначень

$$\tilde{n}_i = n(\vec{r}_i), \quad \tilde{n}_j = n(\vec{r}_i + \vec{a}_j), \quad (4)$$

одержимо для правої частини (4):

$$-\sum_j \nu(\tilde{n}_i - \tilde{n}_j) = -\nu\{n(\vec{r}_i) - n(\vec{r}_i - \vec{a}_x) - n(\vec{r}_i + \vec{a}_x) + n(\vec{r}_i) - n(\vec{r}_i - \vec{a}_y) - n(\vec{r}_i + \vec{a}_y)\} = \nu a^2 \left(\frac{\partial^2}{\partial x_i^2} + \frac{\partial^2}{\partial y_i^2} \right) n(\vec{r}_i) \equiv D \Delta n(\vec{r}_i), \quad (5)$$

Де $D = \nu a^2$ - коефіцієнт дифузії адатомів. Отже рівняння дифузії невзаємодіючих частинок має вигляд:

$$\frac{\partial n(\vec{r}_i, t)}{\partial t} = D \Delta n(\vec{r}_i, t). \quad (6)$$

1.4. Коефіцієнт дифузії взаємодіючих частинок: взаємодія найближчих сусідів

Механізм впливу взаємодії адатомів на ймовірності перескоків пояснено для одновимірного випадку на рис. 1. Якісна картина у двовимірній системі буде такою ж. Якщо у першому випадку енергія активації стрибка з i -го в $i + 1$ - й мінімум зменшується на величину φn_{i-1} , то у двовимірній системі потрібно враховувати вплив усіх найближчих сусідів. Тому зміна енергії активації завдяки а-а взаємодії має вигляд

$$\varphi \sum_j n_j \equiv \varepsilon_i, \quad (7)$$

де, як і раніше, вузли i, j - найближчі сусіди. При активаційному механізмі перескоків

$$\nu_{ij} = \nu_0 e^{-(E_A - \varepsilon_i)} \equiv \nu e^{\varepsilon_i}, \quad (8)$$

Оскільки тепер частота перескоків також залежить від чисел заповнення n_i , то процедура усереднення по часу ускладнюється. Якщо час усереднення дуже великий (прямує до нескінченності), то усереднення фізичних величин по часу еквівалентне усередненню по рівноважному розподілу, заданому функцією

$$\rho = \exp\{\Omega + \mu N_a - E\}, \quad (9)$$

де Ω - великий термодинамічний потенціал, μ - хімічний потенціал (задані в одиницях T), N_a - повна кількість адатомів, E - енергія системи.

$$\tilde{\rho} = \exp\{\tilde{\Omega} + \sum_i \mu_i n_i - E\} \quad (10)$$

$$\delta\rho \approx \rho \left(\tilde{\Omega} - \Omega + \sum_i \delta\mu_i n_i \right). \quad (11)$$

Обидва розподіли ρ і $\tilde{\rho}$ нормовані на одиницю:

$$\sum_{\{n_i\}} \rho = \sum_{\{n_i\}} \tilde{\rho} = 1. \quad (12)$$

Підставляючи знайдену величину Ω в (11), знайдемо значення $d\rho$:

$$\delta\rho = \rho \sum_i \delta\mu_i (n_i - n) \quad (13)$$

$$\delta n_i = \sum_{\{n_k\}} \delta\rho n_i = \sum_{l, \{n_k\}} \rho \delta\mu_l n_i (n_l - n) \equiv \sum_l \delta\mu_l \langle n_i (n_l - n) \rangle. \quad (14)$$

$$\delta n_i = \delta\mu_i \langle \delta N_A^2 \rangle / N = \frac{\partial n}{\partial \mu} \delta\mu_i, \quad (15)$$

Тепер, коли відоме співвідношення між i і nd , можна одержати замкнуте рівняння для еволюції i і nd . З цією метою усереднимо (1) по локально рівноважному розподілу (10). В результаті одержимо

$$\frac{\partial \delta n_{\vec{i}}(t)}{\partial t} = -\nu \sum_{\vec{l}, \vec{j}} \delta \mu_{\vec{l}} \left\langle (n_{\vec{l}} - n) \left(e^{\varepsilon_{\vec{l}}} n_{\vec{i}} h_{\vec{i}+\vec{j}} - e^{\varepsilon_{\vec{l}+\vec{j}}} n_{\vec{i}+\vec{j}} h_{\vec{i}} \right) \right\rangle, \quad (16)$$

де $\vec{l} = \vec{r}_i, \vec{l} = \vec{r}_l, \vec{j} = \vec{a}_j, h_{\vec{k}} = 1 - n_{\vec{k}}$.

У просторово однорідній системі переміщення частинок між найближчими вузлами не залежить від номера вузла та напрямку переміщення. Тоді

$$\left\langle e^{\varepsilon_{\vec{l}}} n_{\vec{i}} h_{\vec{i}+\vec{j}} - e^{\varepsilon_{\vec{l}+\vec{j}}} n_{\vec{i}+\vec{j}} h_{\vec{i}} \right\rangle = 0.$$

Розкладаючи в ряд по малих значеннях a_j з точністю до квадратичних членів і замінюючи $\frac{\partial \mu}{\partial n}$ (див. рівність (15)), одержимо для квадратної решітки рівняння дифузії (15.6), в якому коефіцієнт дифузії дорівнює

$$D = \nu a^2 \frac{\partial \mu}{\partial n} e^{\mu} \langle h_0 h_1 \rangle,$$

де $\langle h_0 h_1 \rangle$ - ймовірність того, що будь-які два сусідні вузли рівноважної системи є незаповненими. Легко переконатись, що значення $\langle h_0 h_1 \rangle$ дорівнює h^2 , коли відсутня взаємодія між адатомами. У випадку притягання між адатомами, наслідком якого є кореляція в заповненні близьких вузлів, значення $\langle h_0 h_1 \rangle$ стає більшим за h^2 . У випадку відштовхування - меншим. Перехід до звичайних одиниць, де хімічний потенціал має розмірність енергії, здійснюється заміною μ на μ/T [3]. Після цього коефіцієнт дифузії набуває вигляду

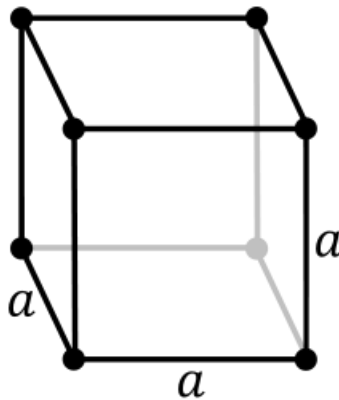
$$D = \nu a^2 \frac{1}{T} \frac{\partial \mu}{\partial n} e^{\mu/T} \langle h_0 h_1 \rangle \equiv \nu a^2 \frac{\partial}{\partial n} (e^{\mu/T}) \langle h_0 h_1 \rangle.$$

Значення усіх фізичних величин, що входять в (17), розраховуються методом Монте Карло.

РОЗДІЛ II. Кристалічні ґратки металів

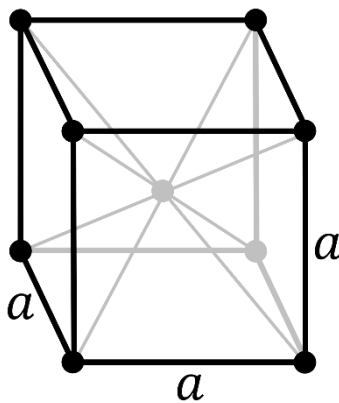
2.1. Основні типи кристалічних решіток металів

Всі метали є кристалічними тілами, що мають певний тип кристалічної решітки, що складається з малорухомих позитивно заряджених іонів, між якими рухаються вільні електрони (так званий електронний газ). Такий тип структури називається металевим зв'язком [4-7].



а)

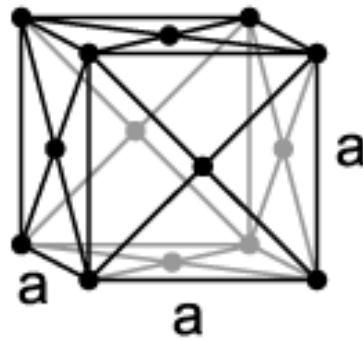
Кубічна (1 атом на комірку)



б)

Об'ємно-центрована кубічна (ОЦК)

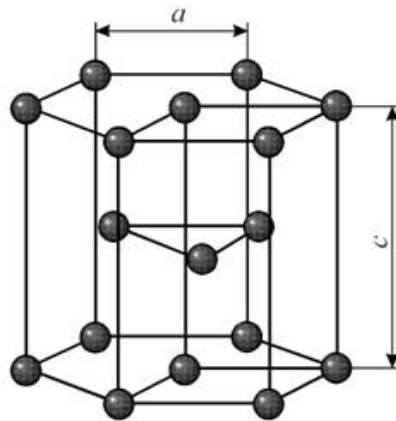
(2 атома на комірку)



в)

Гранецентрирована кубічна (ГЦК)

(4 атома на комірку)



г)

Гексагональна щільноупакована (ГЩ)

(6 атомів на комірку)

Рис.3 Основні типи кристалічних ґраток металів

Метали мають відносно складні типи кубічних ґраток - об'ємно центрована (ОЦК), гранецентрирована (ГЦК) кубічні решітки.

Основу ОЦК-решітки становить елементарна кубічна осередок (рис. 3, б), в якій позитивно заряджені іони металу знаходяться в вершинах куба, і ще один атом в центрі його обсягу. На перетині його діагоналей. Такий тип решітки в певних діапазонах температур мають залізо, хром, ванадій, вольфрам, молібден і інші метали [8].

У ГЦК-решітки (рис.3, в) елементарною клітинкою служить куб з центрованими гранями. Подібну грати мають залізо, алюміній, мідь, нікель, свинець і інші метали.

Третім розповсюдженим різновидом щільноупакованих решіток є гексагональна щільноупакована (ГЦУ, рис. 3, г). ГЦУ-осередок складається з віддалених один від одного на параметр з паралельних зосереджених гексагональних підстав. Три іона (атома) знаходяться на середній площині між підставами [8].

У гексагональних решіток відношення параметра c / a завжди більше одиниці. Таку решітку мають магній, цинк, кадмій, берилій, титан і ін.

Параметр решітки - це відстань між атомами по ребру елементарного осередку. Параметри решітки вимірюються в нанометрів ($1 \text{ нм} = 10^{-9} \text{ м} = 10 \text{ \AA}$). Параметри кубічних грат характеризуються довжиною ребра куба і позначаються літерою a .

Для характеристики гексагональної решітки приймають два параметра - сторону шестикутника a й висоту призми c . Коли відношення $c / a = 1,633$, то атоми упаковані найбільш щільно, і решітка називається гексагональна щільноупакована (рис. 3 г). Деякі метали мають гексагональну решітку з менш щільною упаковкою атомів ($c / a > 1,633$) [8]. Наприклад, для цинку $c / a = 1,86$, для кадмію $c / a = 1,88$.

Параметри a кубічних грат металів знаходяться в межах від 0,286 до 0,607 нм. Для металів з гексагональними гратами a лежить в межах 0,228-0,398 нм, а c в межах 0,357- 0,652 нм.

При підрахунку числа атомів в кожній елементарній комірці слід мати на увазі, що кожен атом входить одночасно в декілька осередків. Наприклад, для ГЦК-решітки, кожен атом, що знаходиться в вершині куба, належить 8 осередків, а атом, що центрує грань, двом. І лише атом, що знаходиться в центрі куба, повністю належить цій комірці.

Таким чином, ОЦК- і ГЦК-осередки містять відповідно 2 і 4 атома.

Під координаційним числом розуміється кількість найближчих сусідів даного атома.

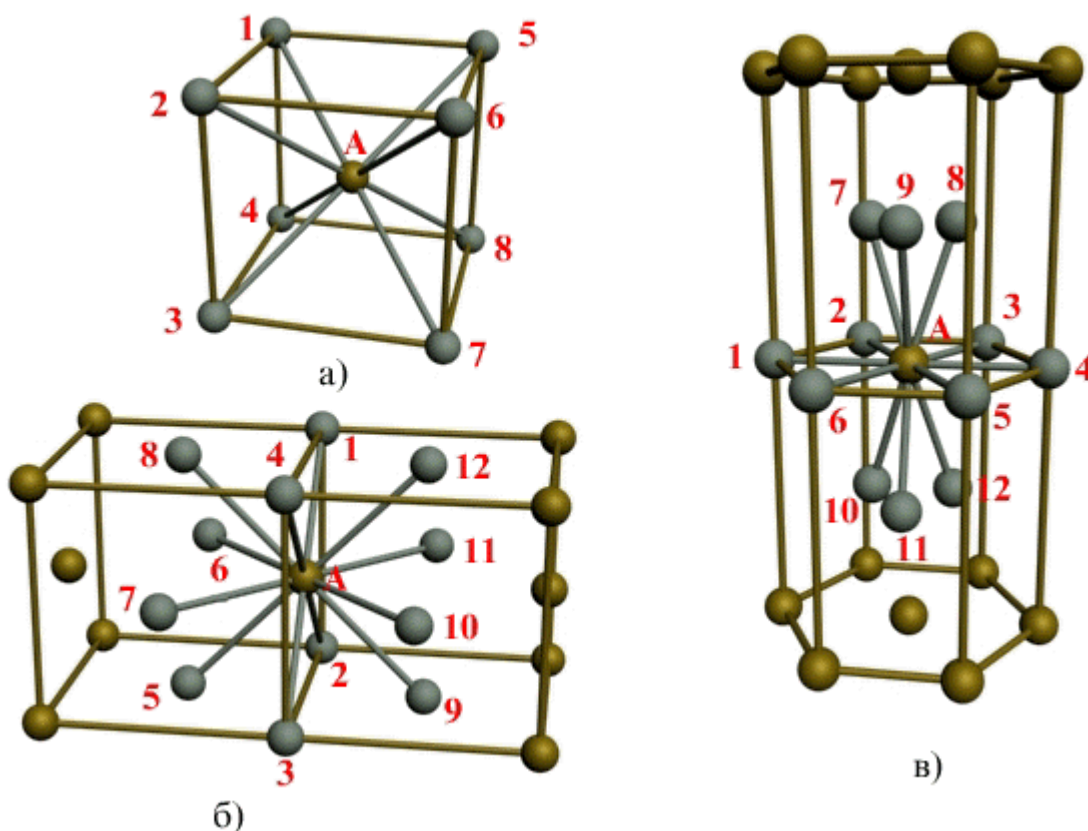


Рис. 4 Координаційне число в різних кристалічних решітках для атома А:
 а) - об'ємноцентрована кубічна (К8); б) - гранецентрирована кубічна (К12); в) - гексагональна щільноупакована (Г12)

В ОЦК решітці (рис. 4, а) атом А (в центрі) знаходиться на найбільш близькому рівній відстані від восьми атомів, розташованих у вершинах куба, тобто координаційне число цієї решітки дорівнює 8 (К8).

У ГЦК решітці (рис. 4, б) атом А (на грані куба) знаходиться на найбільш близькому рівній відстані від чотирьох атомів 1, 2, 3, 4, розташованих у вершинах куба, від чотирьох атомів 5, 6, 7, 8, розташованих на гранях куба, і, крім того, від чотирьох атомів 9, 10, 11, 12, що належать розташованій поруч кристалічній осередку. Атоми 9, 10, 11, 12 симетричні атомам 5, 6, 7, 8. Таким чином, ГЦК решітки координаційне число дорівнює 12 (К12) [9].

У ГПУ решітці при $c/a = 1,633$ (рис. 4, в) атом А в центрі шестигранного підстави призми знаходиться на найбільш близькому рівній відстані від шести атомів 1, 2, 3, 4, 5, 6, розміщених у вершинах шестикутника, і від трьох атомів 7, 8, 9, розташованих у середній площині призми. Крім того, атом А виявляється на такій же відстані ще від трьох атомів 10, 11, 12, що належать кристалічній осередку, що лежить нижче основи. Атоми 10, 11, 12 симетричні атомам 7, 8, 9 [9].

Від типу кристалічної решітки сильно залежать властивості металу.

2.2. Кристалографічні напрямки і площини

Впорядкованість кристалічної будови в просторовій решітці дозволяє виділити окремі кристалографічні напрямки і площини.

Кристалографічні напрямки - це характерні прямі лінії, що виходять з точки відліку, уздовж яких в кристалічній решітці розташовуються атоми. Точками відліку, можуть служити вершини куба, а кристалографічними напрямками - його ребра і діагоналі, а також діагоналі граней (рис. 5, а) [10].

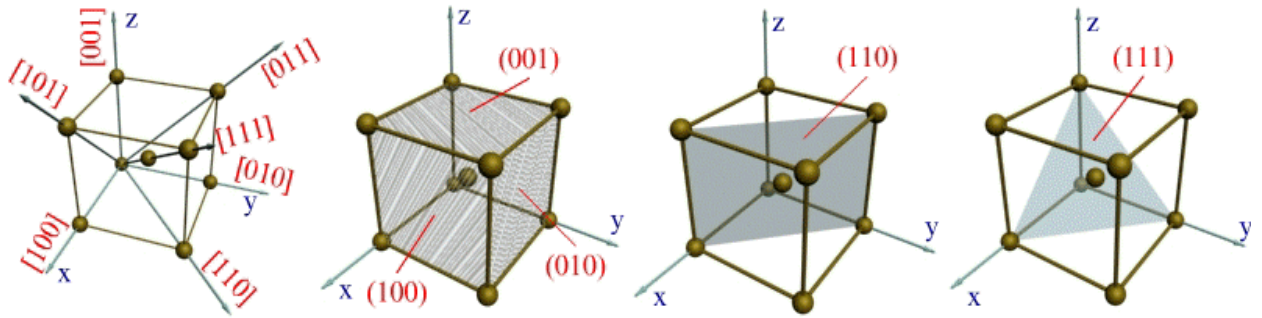


Рис. 5 Кристалографічні напрямки і площини в кристалічній решітці: а) - основні напрямки та їх позначення; б), в), г) - основні площини і їх позначення

Кристалографічними площинами є, наприклад, площини граней кубів (рис. 5, б), а також їх різні діагональні площини разом з розташованими на них атомами (рис. 5, в, г). Для ГПУ-решіток кристалографічними площинами можуть бути площини підстав (рис. 3, г).

Для визначення індексу будь-якого напрямку необхідно знайти індекс найближчого до цієї точки відліку атома, що знаходиться на даному напрямку. Наприклад, індекс найближчого атома уздовж осі ОХ позначається цифрами 100 (рис. 5, а). Ці цифри є координати згаданого атома щодо точки О, виражені через кількість параметрів уздовж осей ОХ, ОУ і ОZ відповідно [10].

Індексом площин, паралельних площинах ХОZ і YOZ, виявляться вираження (010) і (100) (рис. 5, б). Індекс вертикальної діагональної площини куба виразиться через (110), (рис. 5, в), а індекс похилій площині, що перетинає з усіма трьома осями координат на видаленні одного параметра, набуде вигляду (111) (див. рис. 5, г) [10].

РОЗДІЛ III. Моделювання динаміки розпаду вістря вольфрамівих голок

3.1. Опис чисельної моделі

Розглядаємо тривимірну область \mathcal{Q} , в центрі якої розташоване нерухоме ядро (зародок) наночастинки малого розміру (рис.6а). У навколишньому його просторі хаотично рухаються вільні атоми, які можуть приєднуватися до поверхні ядра. Формується наночастинок (НЧ) з збільшуються в часі розмірами внаслідок дифузійних потоків на її поверхню вільних атомів. Розрахунок динаміки всієї системи складається з двох взаємопов'язаних частин:

- моделювання руху ансамблю вільних частинок
- моделювання руху пов'язаних атомів на поверхні наночастинки, яке пределяет еволюцію форми НЧ.

Дифузія вільних атомів. Рух окремих атомів можна описати їх стрибками в випадкових напрямках, які характеризуються випадковим одиничним вектором \vec{e}_f . Довжина стрибка l також є випадковою величиною. Якщо в момент часу t_1 атом знаходився в точці \vec{r}_1 , то в момент $t_2 = t_1 + \Delta t$ він буде в точці $\vec{r}_2 = \vec{r}_1 + l \cdot \vec{e}_f$. Загальна кількість таких «актів розсіювання» при моделюванні одного варіанта завдання доходить до 10^{12} . Якщо випадкові величини l і \vec{e}_f змінюються безперервним чином, то розрахунок зсувів (пов'язаний з обчисленням синусів і косинусів різних кутів) помітно позначиться на загальному часу моделювання. Для істотного скорочення обчислювальної процедури ми перейшли до дискретної моделі розсіювання частинок. Час Δt між окремими актами розсіювання покладається постійним: $\Delta t = \tau$. Довжина стрибків також фіксована: $l = l_0$. Цю величину ми повинні вибирати досить малою. Просторовий розподіл щільності дифузійних потоків вільних атомів $\vec{\Gamma}(\vec{r}, t)$ поблизу поверхні НЧ визначається її формою. Тому

довжина стрибка l_0 у вільному просторі повинна бути порядку міжатомної відстані в наночастинці. Надалі l_0 і τ обрані в якості одиниць вимірювання довжини і часу. У безрозмірних змінних $l = \text{const} = 1, \tau = 1$, обсяг області Q дорівнює $500 \times 500 \times 500$ (розміри вздовж осей координат x, y, z дорівнює 500).

Якщо безперервну множину $\{\vec{e}_f\}$ замінити дискретною $\{\vec{e}_f^k\}$ ($k = 1, 2, 3, \dots, K$), то безліч зсувів $\{l \cdot \vec{e}_f^k\}$ можна обчислити заздалегідь. Потім для кожного акту розсіювання випадковим чином вибирати номер k напрямки розсіювання. Наприклад, використовуючи генератор випадкових чисел \hat{r} , рівномірно розподілених в інтервалі $(0, 1)$. В цьому випадку $k = \text{int}(\hat{r} \cdot K) + 1$ при $\hat{r} < 1$, де $\text{int}(\dots)$ - ціла частина числа; $k = K$ якщо $\hat{r} = 1$. По знайденому номеру k витягаємо з пам'яті комп'ютера вже обчислені величини $l \cdot \vec{e}_f^k$. Таким чином, ми уникаємо багаторазових обчислень синусів і косинусів, а час обчислень не залежить від параметра дискретизації K .

Методика формування $\{\vec{e}_f^k\}$ наступна. Неперервна множина $\{\vec{e}_f\}$ рівномірно заповнює сферу одиничного радіуса. У сферичній системі координат кожен випадковий вектор \vec{e}_f задається кутом θ (θ -кут між вектором \vec{e}_f та вісю Z) і азимутом φ . Розподіл щільності ймовірності $p(\varphi)$ кута φ рівномірне в інтервалі $0 \leq \varphi \leq 2\pi$, тому дискретна множина $\{\varphi_i\}$ ($i = 1, 2, \dots, N_\varphi$) можна сформувати згідно співвідношенню $\varphi_i = \frac{2\pi}{N_\varphi}(i-1)$. Розподіл щільності ймовірності $p(\theta)$ кута θ в інтервалі $0 \leq \theta \leq \pi$ *неравномірне*: $p(\theta) = \frac{1}{2} \sin \theta$.

Отже, випадковий кут θ задовольняє співвідношенню $\frac{1}{2} \int_0^\theta \sin \theta' d\theta' \equiv \frac{1}{2} (1 - \cos \theta) = \hat{r}$, де \hat{r} - випадкова величина, рівномірно розподілена в інтервалі $(0, 1)$. Це співвідношення використовуємо для формування дискретної множини $\{\theta_j\}$ ($j = 1, 2, \dots, N_\theta$). Проміжок $(0, 1)$ розбиваємо на N_θ рівних частин. Кожній частині з

номером j зіставляємо кут $\theta_j = \arccos\left(1 - \frac{2}{N_\theta}(j - 0.5)\right)$. При такій дискретизації кожен кут θ_j равновероятен в акті розсіювання вільного атома, але розподіл всіх значень θ_j в інтервалі $(0, \pi)$ нерівномірно. Всілякі комбінації кутів φ_i і θ_j представляють шукану множину векторів $\{\vec{e}_f^k\}$ ($k = i \cdot j$). При цьому ймовірності розсіювання уздовж будь-якого напрямку \vec{e}_f^k однакова.

У нашій моделі використані значення $N_\theta = 20$, $N_\varphi = 40$. Еквівалентність дискретної множини $\{\vec{e}_f^k\}$ та безперервної $\{\vec{e}_f\}$ легко проконтролювати. Побудуємо послідовність M вектора \vec{e}_m ($M \ll K = N_\theta N_\varphi$), випадково обираємо з множини $\{\vec{e}_f^k\}$. Для довільного одиничного вектора \vec{e}_0 середнє значення

$$\langle s_1^2 \rangle \equiv \langle (\vec{e}_0 \cdot \vec{e}_m)^2 \rangle = \frac{1}{M} \sum_{m=1}^M (\vec{e}_0 \cdot \vec{e}_m)^2 \quad \text{повинно} \quad \text{бути} \quad \text{близько} \quad \text{до}$$

$$\langle (\vec{e}_0 \cdot \vec{e}_f)^2 \rangle = \langle (\vec{e}_f)^2 \rangle = \frac{1}{4\pi} \int_0^{2\pi} d\varphi \int_0^\pi \cos^2 \theta \sin \theta d\theta = 1/3. \quad \text{У} \quad \text{наших} \quad \text{розрахунках}$$

$\max[\langle s_1^2(\vec{e}_0) \rangle - 1/3] = 8 \times 10^{-4}$. Величина $\langle s_1^2 \rangle$ при фіксованій довжині стрибка $l = 1$ є середньоквадратичним зміщенням атома в обраному напрямку за час $t = 1$. Саме цей параметр $\langle s_1^2 \rangle$ - визначає коефіцієнт дифузії D вільних атомів.

Відзначимо, що ми відмовилися від опису дифузії рівнянням, отриманим в наближенні суцільного середовища: $\frac{\partial n(\vec{r})}{\partial t} = D \cdot \Delta n(\vec{r})$ ($n(\vec{r})$ - концентрація атомів), з двох причин. Перша з них - складність визначення граничної умови для концентрації $n(\vec{r})$ на поверхні наночастинки. Друга обумовлена тим, що в ряді випадків розмір наночастинки можна порівняти із середнім відстанню між вільними атомами, особливо в початковій стадії її зростання.

Коефіцієнт дифузії може бути легко визначений із співвідношення: $D = \langle s^2 \rangle (2t)^{-1}$, де $\langle s^2 \rangle$ - середнє квадратичне зміщення вільного атома в довільно

виділеному напрямку за час t . Без порушення загальності t можна вибрати рівним $\tau=1$. Тоді

$$D = \frac{1}{2} \frac{\langle s_1^2 \rangle}{\tau} = 1/6. \quad (18)$$

Ми фіксували довжину стрибка вільного атома l і інтервал часу між стрибками Δt . Таке спрощення процесу дифузії вільних атомів не позначається на кінцевих результатах для всієї системи. Якщо величини l і Δt приймати випадковими функціями часу, то вид цих функцій визначить коефіцієнт дифузії D , і він буде відрізнятися від $1/6$. Однак природно припустити, що динаміка системи залежить від добутку $D \times n_0(t)$, а не від величини кожного параметра окремо ($n_0(t)$ -концентрація атомів на кордоні розрахункової області). При розрахунку різних варіантів завдання концентрація $n_0(t)$ підтримується нами на постійному рівні, заданому в момент часу $t=0$. Для цього періодично підраховується число атомів N_b в трьох прикордонних шарах області Q , загальний обсяг яких $V_b \approx 4.45 \times 10^6$. Якщо N_b менше заданого значення N_{b0} , то в прикордонні шари додають $(N_{b0} - N_b)$ вільних атомів із випадково обраними координатами. При $n_0 \approx 1.8 \times 10^{-3}$. Тобто поблизу кордону один вільний атом міститься в кубі зі стороною $b \approx 8$.

Відзначимо, що дрейф атома від кордону області Q до її центральної частини (пройдений шлях $L \approx 200$) триває час $t_d \approx L^2 / (2D) = 1.2 \times 10^5$, і стільки ж зіткнень він відчуває.

Моделювання динаміки пов'язаних атомів на поверхні наночастинки. Припускаємо, що атоми ядра, число яких ~ 10000 , розташовуються у вузлах кристалічної решітки певного типу. Атоми, які осідають на ядро в процесі дифузії, добудовують задану просторову структуру, розташовуючись в найближчих до ядра вакантних вузлах (рис. 6а - імовірнісні закони, що керують

динаміків пов'язаних атомів). Відзначимо, що чисельна модель не розглядає еволюцію ядра від моменту зародження і механізм формування решітки на ранніх етапах його утворення. Це окрема складна проблема. Наша мета - вивчення закономірностей більш пізніх етапів росту. В роботі розглянуті випадки простий кубічної, кубічної об'ємно-центрованої, кубічної гранецентрированої і гексагональної плотноупакованої решіток. Відстань між сусідніми атомами в гексагональній решітці вважається рівним одиниці. Постійна решітки (a) в простій кубічній і кубічній об'ємно-центрованої решітках також прийнята за одиницю ($a=1$), для кубічної гранецентрированої решітки $a=2l=2$. Відзначимо, що довжина стрибків вільних атомів не перевищує постійної решітки першій-ліпшій нагоді, що дозволяє цим атомам дифундувати в область каверн, які можуть утворюватися на поверхні зростаючої НЧ. Ядро в більшості варіантів задано безліччю атомів решітки, що лежать всередині сфери радіуса $r_0=15$. Такий вибір заснований на результатах розрахунків, коли ядро (чисто умовно) представляло ланцюжок з 11 жорстко пов'язаних атомів. Навіть цьому випадку, починаючи з загального числа атомів ~ 400 , що росте нанокристал набував сфероїдальної вид, зберігаючи його до моменту досягнення радіусу $r(t) \approx 15$. Таку НЧ для скорочення часу розрахунків можна спочатку поставити в якості ядра.

Якщо атоми, що приєднуються до ядра, вважати нерухомими, то в цьому випадку, як відомо, зростає фрактал незалежно від щільності дифузійного потоку вільних атомів. В реальності приповерхневих атоми можуть перекочуватися по поверхні зростаючої НЧ (мінімізуючи її потенціальну енергію) і відриватися від поверхні під дією випадкових збурень, що призводить до зростання компактних НЧ. Зазначені процеси враховані в чисельній моделі наступним чином. З інтервалом $\Delta t = \tau = 1$ аналізуємо можливість зміни положення кожним з атомів, що входять до складу НЧ (за винятком «внутрішніх» атомів, які не мають в своєму оточенні вакансій в кристалічній решітці). Обчислювальну процедуру для простоти

розглянемо на прикладі двовимірної системи (рис.6 б). Вибираємо один з поверхневих атомів (наприклад, атом а).

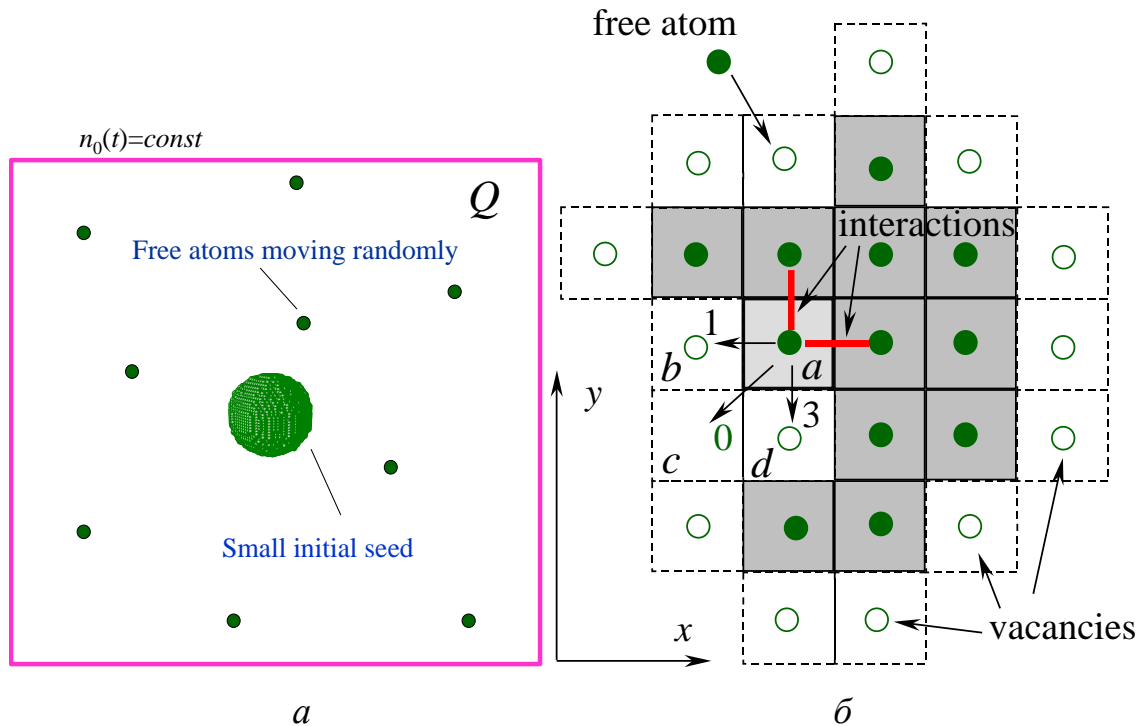


Рис. 6 а-Геометрія системи. б -Ілюстрація до схеми розрахунку динаміки поверхневих атомів НЧ в двовимірній системі. Атоми НЧ розташовані в вузлах квадратної решітки (центри тонованих квадратів). Передбачається, що кожен атом може взаємодіяти з найближчими сусідами по напрямках типу (1,0), тобто максимальне число зв'язків дорівнює 4. Енергія парного взаємодії дорівнює (). Числа в осередках вказують на число атомів, з якими буде взаємодіяти атом осередку, якщо перейде в одну з цих осередків. Вільний атом захоплюється поверхнею НЧ, якщо відстань від нього до найближчої приповерхностной вакансії стає менше $\frac{1}{2}$ (половини постійної решітки).

1. Спочатку визначаємо ймовірність того ($P_{a,mov}$) що атом a може змінити своє положення: $P_{a,mov} = (p_0)^{k_a}$, де p_0 - параметр задачі, k_a - число сусідніх атомів з якими взаємодіє атом a ($k_a=2$, рис. 1b; в даному випадку реалізовані дві взаємодії з чотирьох можливих для внутрішнього атома

НЧ). Генеруємо випадкове число r ($0 \leq r \leq 1$). Якщо $r > P_{a,mov}$, то атом a залишаємо нерухомим і переходимо до розрахунку динаміки наступного атома.

2. Якщо $r \leq P_{a,mov}$, то атом a може здійснити перехід в комірку b або d , «випаруватися» з поверхні НЧ (комірка c), або, все-таки, залишитися в попередньому положенні (комірка a). У новому положенні b атом a буде пов'язаний з одним із сусідніх атомів ($k_b = 1$). Для нового положення d $k_d = 3$; $k_c = 0$ у положенні c . Імовірність p_ξ кожного з чотирьох положень ξ ($\xi = a, b, c, d$), в якому може опинитися атом a , визначаємо в моделі термостата:

$$p_\xi = C \times \exp\left(-\frac{\varepsilon k_\xi}{T}\right), \quad \sum_{\xi=a}^d p_\xi = 1. \quad (19)$$

Тут T - температура, C - нормувальний множник.

Після обчислення ймовірностей p_ξ генеруємо випадкове число $\hat{r} \in (0,1)$ і визначаємо індекс ξ_{mov} нового положення атома з умови:

$$S(\xi_{mov}) = \sum_{\xi=a}^{\xi_{mov}} p_\xi \geq \hat{r}, \quad (20)$$

де $S(\xi_{mov})$ - мінімальна часткова сума, що перевищує величину \hat{r} .

У тривимірній системі числення процедура розрахунку динаміки поверхні НЧ аналогічна. Для вирішення завдання повинні бути визначені:

- Тип кристалічної решітки ядра (наночастинки);
- Напрямки $\{\vec{e}_{int}\}$, по которым атоми НЧ взаємодіють з сусідніми атомами. за якими атоми НЧ взаємодіють з сусідніми

атомами. Наприклад, для простої кубічної решітки розглядаємо шість напрямків взаємодії типу $(1,0,0)$.

- Напрямки $\{\vec{e}_{mov}\}$, за якими атоми НЧ можуть переходити в сусідні вакансії. Множина $\{\vec{e}_{mov}\}$ в загальному випадку ширше множини $\{\vec{e}_{int}\}$. Очевидно, що в простий кубічної гратці приповерхневий атом НЧ може здійснювати стрибки в найближчі вакансії уздовж напрямків $\{\vec{e}_{int}\}$. Однак можна припустити і стрибки в сусідні вакансії уздовж 12 діагональних напрямків виду $(1,1,0)$.
- Параметри p_0 і $\alpha = \varepsilon/T$. Число атомів N_{b_0} (концентрацію вільних атомів на кордоні розрахункової області n_0).

Параметр $\alpha = \varepsilon/T$ вибирається в такий спосіб. У початковий момент часу в центрі розрахункової області \mathcal{Q} поміщається наночастинка сфероїдальної форми з відносно великим характерним розміром $R \approx 50$. Розраховується її еволюція в часі за відсутності потоку вільних частинок від кордонів \mathcal{Q} для довільного значення α . Якщо цей параметр виявляється менше критичного значення, $\alpha < \alpha_{cr}$, то наночастинка повністю випаровується. Якщо $\alpha > \alpha_{cr}$, то з часом встановлюється рівновага частки з навколишньою насиченою «парою» низької щільності (на два порядки нижче типовою граничної концентрації $n_0 = 1.8 \times 10^{-3}$). Форма частки в рівноважному стані відповідає конфігурації з мінімальною поверхневою енергією. Параметр p_0 головним чином визначає час встановлення рівноваги, але ε_{cr} слабо залежить від p_0 .

Для вирішенні завдання в повному обсязі ми вибираємо α вище критичного на 30-50%, оскільки радіус ядра $r_0 \approx 15$, і «випаровування» з його поверхні йде інтенсивніше. Крім того, вибір параметра α «із запасом»

необхідний для утворення гранованою форми НЧ. В іншому випадку, вона мало відрізняється від сферичної.

3.2. Розпад вістря вольфрамкових голок на краплі

В роботі представлені результати моделювання динаміки розпаду вольфрамкових голок в припущенні, що всі атоми (внутрішні та поверхневі) розташовані у вузлах ОЦК ґратки (постійна ґратки a , прийнята за одиницю виміру довжини). Поверхневі атоми можуть дрейфувати уздовж поверхні вольфрамкових голок. Напрямки стрибків у сусідні вакансії решітки мають різні ймовірності в залежності від числа зв'язків в нових станах. Закладена також можливість сублімації атомів - відрив від поверхні голки з наступною дифузією в навколишньому просторі і, відповідно, reattachment. Розглянута динаміка розпаду голок вирізаних вздовж кристалографічних осей (100). Представлено, що для такої моделі реалізується розпад голок внаслідок поверхневої дифузії атомів. Можливе зменшення числа пов'язаних атомів в часі через акти detachment в процесі розпаду виключено зануренням наноголок в капсулу, стінки якої відбивають вільні атоми. Відстань між стінками капсули і поверхнею наноголки оптимізовано. З одного боку, вони не повинні впливати на досліджувані процеси і бути досить віддаленими, а з іншого - не допускати значного зменшення кількості зв'язаних атомів, N_t , в квазірівноважному стані нанокластер-пар щодо їх початкового значення, N_0 . Ця рівновага встановлюється досить швидко, однак може супроводжуватися значною перебудовою структури поверхні. Детально фізичний зміст параметрів моделі α і p описаний в розділі 3. Зараз відзначимо лише, що $\alpha \sim 1/T$, і менші значення α відповідають більшій температурі.

На рис.7 представлена початкова форма наноголок, яка являє собою усічений конус з радіусами $r_1 = 16a$, $r_2 = 4a$, до початку процесу дифузії.

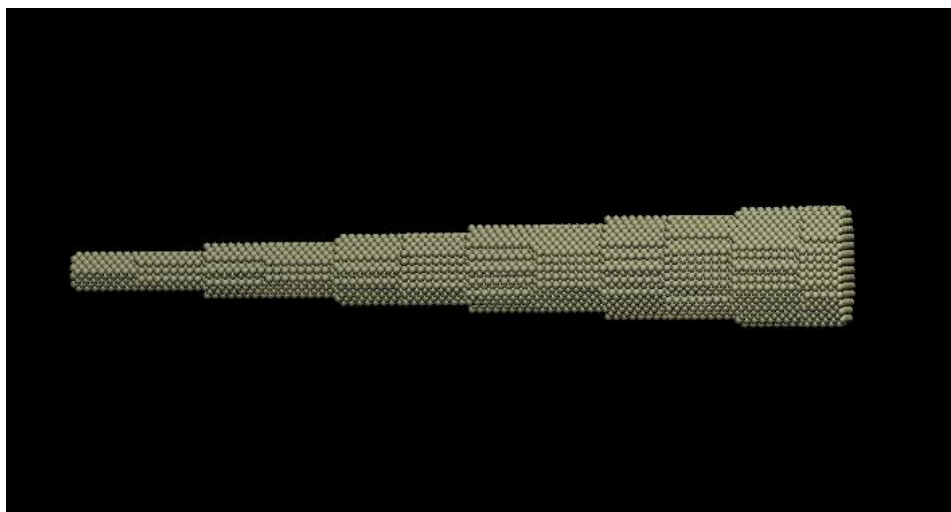
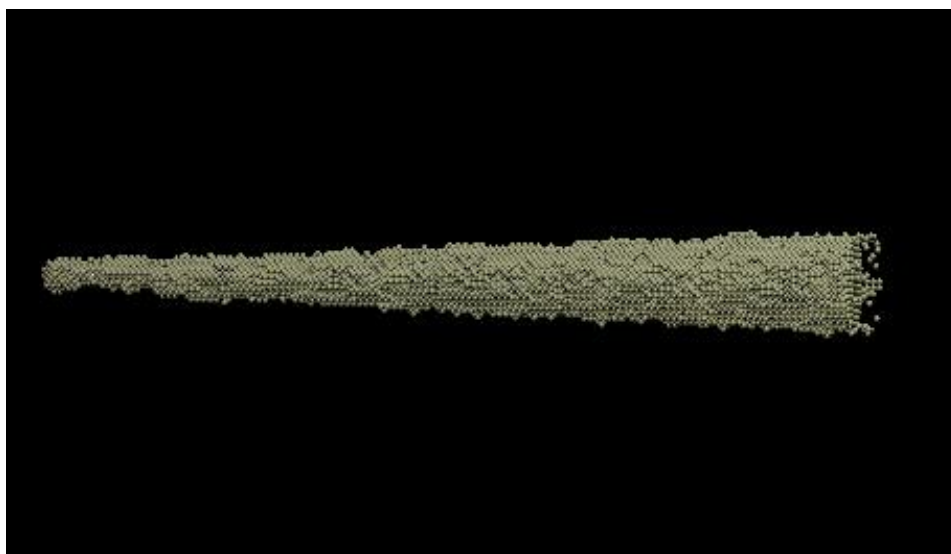
*a**b*

Рис.7 Початкова форма наноголки до (а) та після (b) початку процесу дифузії. Холодний режим - $\alpha = 1.5$, $p = 0.9$, $r_1 = 16a$, $r_2 = 4a$, $L = 275a$, $N_0 = 20734$. Сім нижніх слоїв заморожені, еволюція визначається тільки процесами на вістрі голки. b - $N_t = 20600$, $t = 0.01 \times 10^6$.

На рис. 7-б бачимо, що процес дифузії вже почався ($t = 0.01 \times 10^6$). Форма наноголки потроху змінюється. Біля вістря голки, з одного боку починає утворюватися перетяжка, яка в подальшому може призвести до утворення краплі.

При орієнтації nanorod вздовж напрямку (100) бокова поверхня обмежена чотирма площинами (100) і чотирма площинами (110). У квазірівноважному стані структура поверхні досить хаотична, але фрагменти площин (100) значною мірою зберігаються. В результаті поверхневої дифузії

вістря наноголки (зони підвищеного середнього поверхневого тиску) потовщуються - звертаємо на цей факт особливу увагу рис. 8 (a-b). Значною мірою bulbous ends обмежені фрагментами рівноважної конфігурації Вульфа. Утворення bulbous ends (потовщення на кінці вістря голки) призводить до відтоку атомів від центра голки, та призводить до утворення перетяжки.

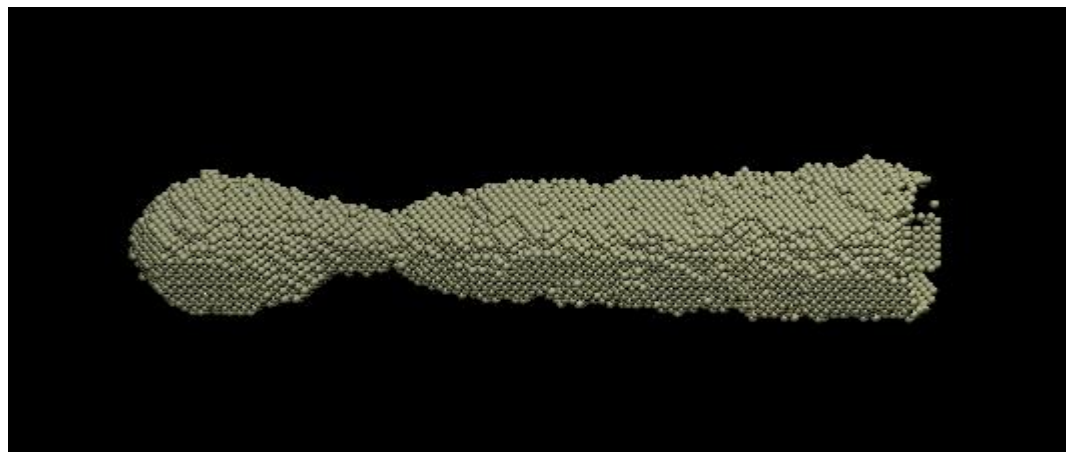
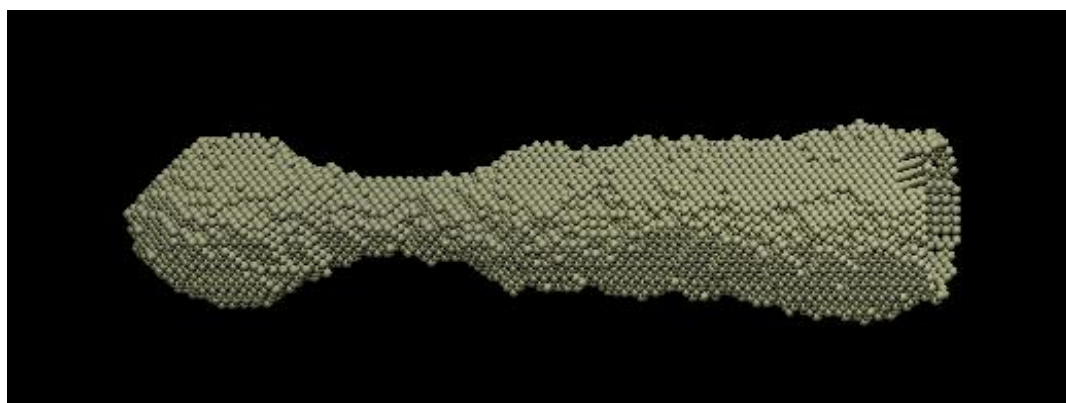
*a**b*

Рис 8. Утворення bulbous end на кінці вістря голки:

a) $t = 3 \times 10^6$, $N_t = 20481$.

b) $t = 3.5 \times 10^6$, $N_t = 20431$.

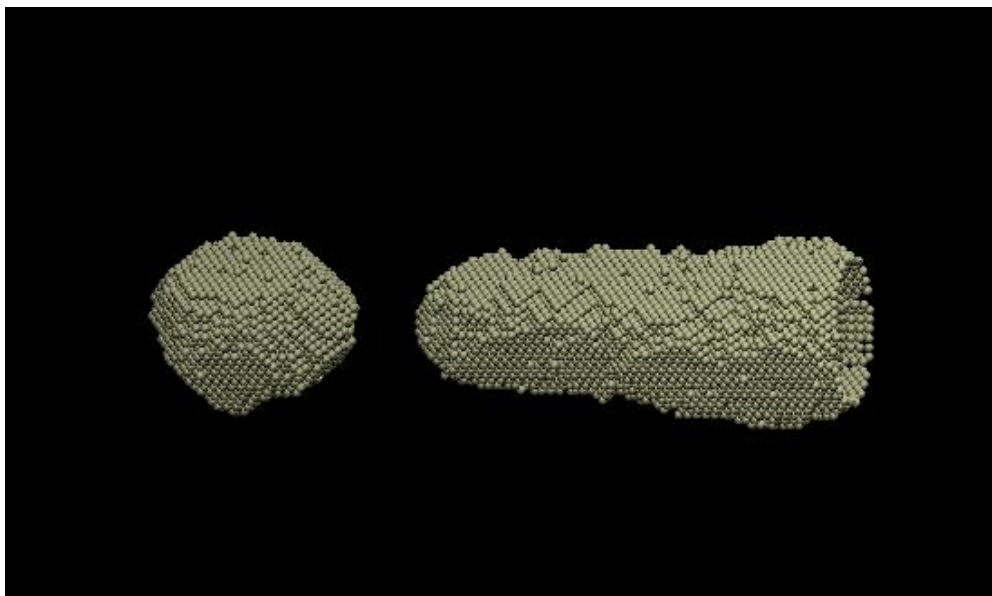


Рис. 9 Утворення краплі. Холодний режим $\alpha = 1.5$, $p = 0.9$, $r_1 = 16a$, $r_2 = 4a$, $N_0 = 20734$, $N_t = 20460$, $t = 4 \times 10^6$. Сім нижніх атомів заморожені.

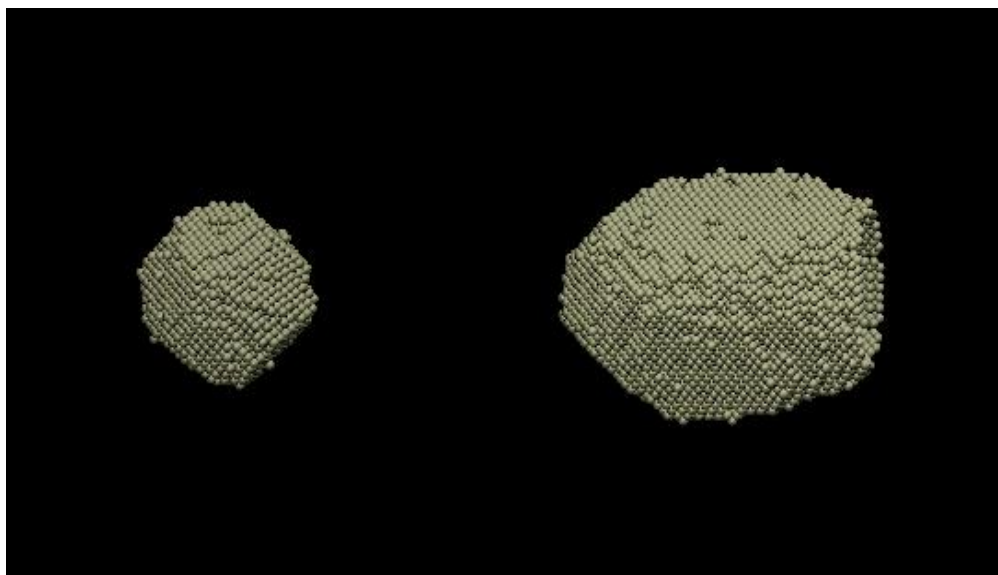


Рис. 10 Утворення краплі, подальший розпад не відбувається. Холодний режим $\alpha = 1.5$, $p = 0.9$, $r_1 = 16a$, $r_2 = 4a$, $N_0 = 20734$, $N_t = 20440$, $t = 9 \times 10^6$. Сім нижніх атомів заморожені.

Як бачимо з рис.10 подальший розпад наноголки на краплі не відбувся. При подальшому розрахунку краплі, що відокремилися від наноголки приймають форму близьку до рівноважної (рис. 9-10), навіть при відносно малому числі атомів в них. Інша частина наноголки, що залишилася після утворення краплі, починає випаровуватися та заокруглюється.

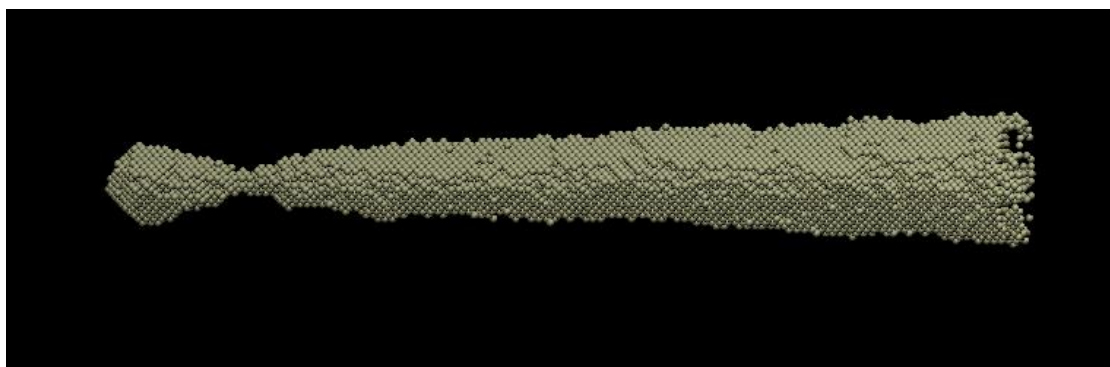
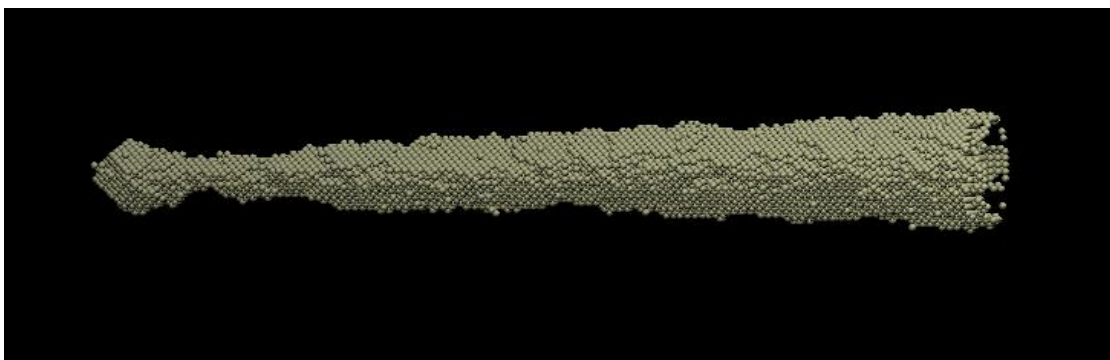
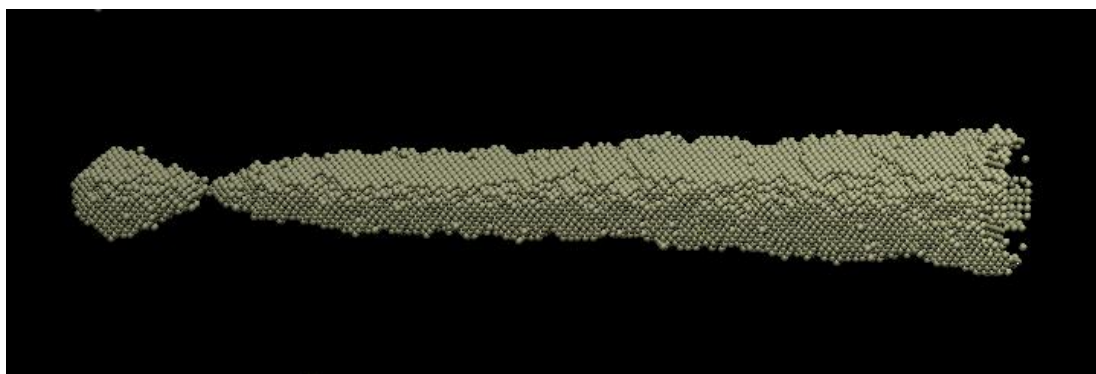
*a**b**c*

Рис 11. Утворення bulbous end на кінці вістря голки (без сублімації):

a) $t = 0.45 \times 10^6$, $N_t = 20734$. b) $t = 0.53 \times 10^6$, $N_t = 20734$ c) $t = 0.538 \times 10^6$, $N_t = 20734$

Розглянемо розпад вістря вольфрамівих голок при відключеному процесі сублімації (рис. 11-14). Початкову форма та всі параметри залишаймо ті самі: холодний режим - $\alpha = 1.5$, $p = 0.9$, $r_1 = 16a$, $r_2 = 4a$, $N_0 = 20734$, сім нижніх атомів заморожені.

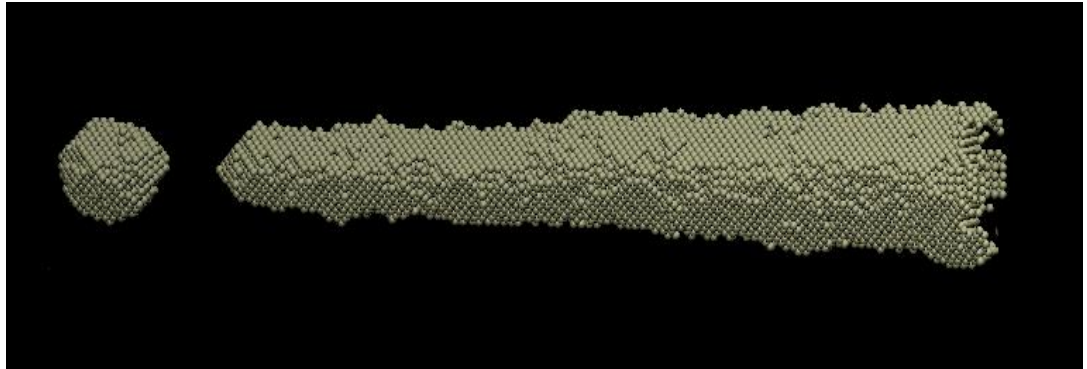


Рис 12. Утворення краплі при відключеному процесі сублімації.

$t = 0.57 \times 10^6$, $N_t = 20734$, $L = 275a$.

Зазначемо, що утворення bulbous ends (рис. 11) та краплі (рис. 12) на кінці вістря голки, при вимкненому процесі сублімації проходить значно швидше ніж в попередніх випадках (рис. 8-10).

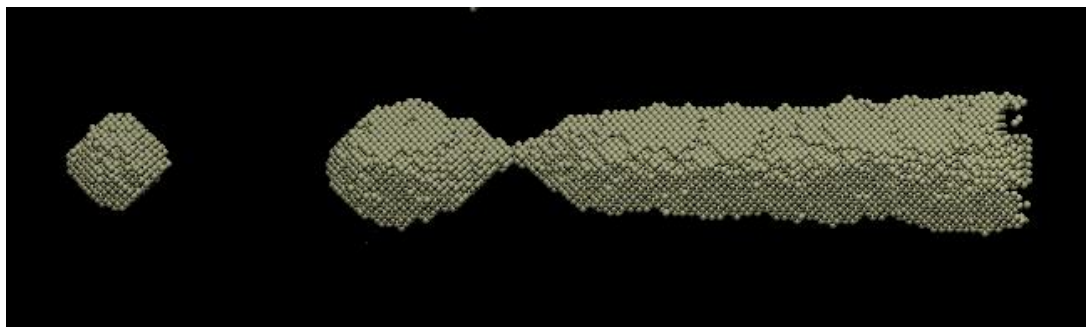
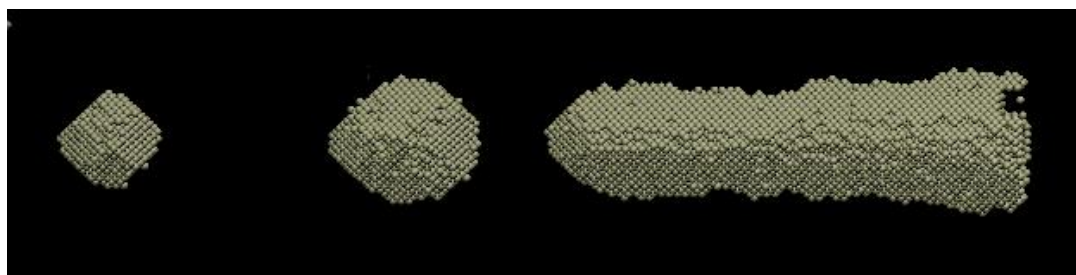
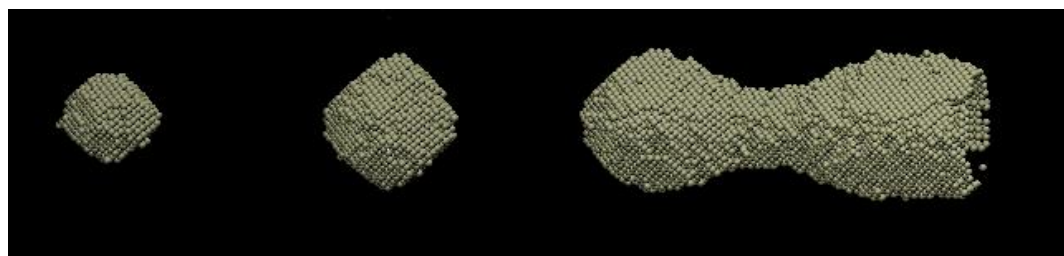


Рис 13. Утворення краплі при відключеному процесі сублімації. $t = 1.7 \times 10^6$, $N_t = 20734$, $L = 275a$.



a



b

Рис 14. Утворення другої краплі при відключеному процесі сублімації. $t = 2 \times 10^6, 4 \times 10^6, N_t = 20734, L = 275a$.

Слід зазначити, що в такому режимі розпаду («відщипування» крапель від кінців) генерується в середньому більше число крапель, ніж з «включеної» сублімацією.

Взагалі, сублімація «направляє» систему по траєкторії, на фініші якої система буде володіти меншою потенційною енергією, ніж в передбачуваному процесі БЕЗ сублімації.

ВИСНОВКИ

У роботі представлена комп'ютерна модель розпаду вістря вольфрамівих голок на краплі. Для реалізації програми розглянуто процеси поверхневої дифузії. Розроблена тривимірна модель дифузійного росту наночастинок, внутрішня будова яких відповідає ОЦК ґратці. Врахована динаміка поверхневих частинок кластера, які з різним ступенем ймовірності можуть змінювати своє положення при переході в сусідні вакансії решітки або відриватися від поверхні.

Отримані результати свідчать, що відрив краплі від вістря наноголки відбувається завдяки поверхневому потоку атомів. Цей потік прямує від вістря наноголки до її основи, та призводить до формування потовщення у верхній частині. Відриву краплі від поверхні наноголки може відбуватися тільки в разі, якщо суміжні поверхні (що лежать по різні боки від максимальної претяжки) утворюють досить тупий кут, що мінімізує прояв негативної кривизни перетяжки в площині, що проходить через вісь системи.

Отримана залежність розпаду вістря вольфрамівих голок від їх довжини та кута вістря.

Розпад вістря голки на краплі при вимкненому процесі сублімації генерує в середньому більшу кількість крапель, за менший час, ніж з сублімацією.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. D. T. Robb and V. Privman, *Langmuir* 24, 26 (2008); V. Privman, *Ann. NY Acad. Sci.*, in print (arxiv.org/abs/0707.2595) and *J. Optoelectr. Adv. Mater.*, in print (arxiv.org/abs/0806.4644).
2. A.A. Chumak and A.A. Tarasenko, Diffusion and density fluctuations of atoms adsorbed on solid surface, *Surface Science*, 91 (1980) 694-706.
3. А.А. Тарасенко, П.М. Томчук, А.А. Чумак, Флуктуации в объеме и на поверхности твердых тел, “Наукова думка”, 251 с., Киев, 1992.
4. C. M. Wang, D. R. Baer, J. E. Amonette, M. H. Engelhard, Y. Qiang and J. Antony, *Nanotechn.* 18, 255603 (2007).
5. J. Pan and R. Huang, *Key Eng. Mater.*, 2008, pp.368–372.
6. Lord Rayleigh, On the instability of jets, *Proc. London Math. Soc.* 10, 4 (1878). <https://doi.org/10.1112/plms/s1-10.1.4>
7. S Karim, M E Toimil-Molares, A G Balogh, W Ensinger, T W Cornelius, E UKhan and R Neumann, Morphological evolution of Au nanowires controlled by Rayleigh instability, *Nanotechnology* 17 (2006) 5954–5959, [doi:10.1088/0957-4484/17/24/009](https://doi.org/10.1088/0957-4484/17/24/009).
8. C. Bréchnignac, Ph. Cahuzac, F. Carlier, C. Colliex, J. Leroux, A. Masson, B. Yoon, and Uzi Landman, Instability Driven Fragmentation of Nanoscale Fractal Islands, *Phys. Rev. Lett.* 88, NUMBER 19, 13 MAY 2002, [doi: 10.1103/PhysRevLett.88.196103](https://doi.org/10.1103/PhysRevLett.88.196103).
9. Donghong Min and Harris Wong, **Rayleigh’s instability of Lennard-Jones liquid nanothreads simulated by molecular Dynamics**, *Phys. Fluids* 18, 024103 (2006); [doi: http://dx.doi.org/10.1063/1.2173620](http://dx.doi.org/10.1063/1.2173620).
10. Amlan Dutta, Swastika Chatterjee, A. K. Raychaudhuri, Amitava Moitra, and T. Saha-Dasgupta, **In-silico investigation of Rayleigh instability in ultra-thin copper nanowire in premelting regime**, *J. Appl. Phys.* 115, 244303 (2014); <http://dx.doi.org/10.1063/1.4885044>.

11. M. Tjahjadi, H. A. Stone and J. M. Ottino, **Satellite and subsatellite formation in capillary Breakup**, J. Fluid Mech. (1992), v. 243, pp. 297-317, doi: <https://doi.org/10.1017/S0022112092002738>.
12. H. A. Stone, **Dynamics of drop deformation and breakup in viscous fluids**, Annual Review of Fluid Mechanics, (1994), vol. 26, pp.65-102, DOI:10.1146/annurev.fl.26.010194.000433.
13. V. N. Gorshkov and D. V. Mozyrskii, **Self-excitation of short-wave structures and breakup into droplets in bounded liquid filaments**, Technical Physics, Volume 41, Issue 10, October 1996, Pages 980-985.
14. R. M. S. M. Schulkes, The contraction of liquid filaments, J. Fluid Mech., Volume 309 February 1996, pp. 277-300, doi: <https://doi.org/10.1017/S0022112096001632>.
15. Jens Eggers, **Nonlinear dynamics and breakup of free-surface flows**, Reviews of Modern Physics, Vol. 69, No. 3, July 1997, pp. 865-929.
16. V. N. Gorshkov and M. G. Chaban, **Nonlinear electrohydrodynamic phenomena and droplet generation in charged jets of conducting liquid**, Technical Physics, v. 44, Number 11 November 1999, pp. 1259-1266, doi:10.1134/1.1259506.
17. A.A. Castrejón-Pita, J.R. Castrejón-Pita, and I.M. Hutchings, **The breakup of liquid filaments**, Phys. Rev. Lett. 108, 074506 (2012).
18. C.-H. Zhang, F. Kassubek, and C. A. Stafford, **Surface fluctuations and the stability of metal nanowires**, Phys. Rev. B 68, (2003).
19. Sevonkaev, I.; Goia, D. V.; Matijevic, E. J. Colloid Interface Sci. 2008, 317, 130.
20. Halaciuga, I.; Goia, D. V. J. Mater. Res. 2008, 23, 1776.
21. Goia, D. V.; Jitianu, M. J. Colloid Interface Sci. 2007, 309, 78.
22. Jitianu, M.; Kleisinger, R.; Lopez, M.; Goia, D. V. J. New Mater. Electrochem. Syst. 2007, 10, 67.
23. Wang, C. M.; Baer, D. R.; Amonette, J. E.; Engelhard, M. H.; Qiang, Y.; Antony, J. Nanotechnology 2007, 18, 255603.

24. Matijevic, E. *Colloid J.* 2007, 69, 29.
25. Uskokovic, V.; Matijevic, E. *J. Colloid Interface Sci.* 2007, 315, 500.
26. Nair, P. S.; Fritz, K. P.; Scholes, G. D. *Small* 2007, 3, 481.
27. Embden, van J.; Jasieniak, J.; Gomez, D. E.; Mulvaney, P.; Giersig, M. *Aust. J. Chem.* 2007, 60, 457.
28. Yin, Y. D.; Erdonmez, C.; Aloni, S.; Alivisatos, A. P. *J. Am. Chem. Soc.* 2006, 128, 12671.
29. Zeng, H. B.; Liu, P. S.; Cai, W. P.; Cao, X. L.; Yang, S. K. *Cryst. Growth Des.* 2007, 7, 1092.

ДОДАТКИ

Основний код програми.

```
using System;
using System.IO;
namespace Молекулки
{
    class Grid
    {
        int R = 2,
            x0 = 5,
            y0 = 5,
            z0 = 5;

        public int
            dimX = 10,
            dimY = 10,
            dimZ = 10;
        // public int d;
        public double koevr = 0.9;

        public int[,.] volume;

        public Grid()
        {
            volume = new int[dimX, dimY, dimZ];
            // volume[2, 4, 1] = 10;
        }

        public void Output()
```

```

{

    for(int i = 0; i < dimX; i++)
    {
        for(int j = 0; j < dimY; j++)
        {
            for (int k = 0; k < dimZ; k++)
            {
                if (volume[i, j, k] != 0)
                {
                    Console.Write(" X ");
                }
                else
                {
                    Console.Write(" 0 ");
                }
            }
        }
        Console.WriteLine("|");
    }
}

public void MakeRod()
{
    for (int i = 0; i < dimX; i++)
    {
        for (int j = 0; j < dimY; j++)
        {
            for (int k = 0; k < dimZ; k++)
            {
                //volume[250, 250, 1] + 2 * Math.PI * R = 1;
            }
        }
    }
}

```

```

        if (isRod(i, j, k) && isMatched(i, j, k))
        {
            volume[i, j, k] = 100;
        }
        else
        {
            volume[i, j, k] = 0;
        }
    }
}
}
}

```

```

public Boolean isRod(int x, int y,int z)
{
    // int a = 1, b = 1, c = 1;
    //d = (a*x+b*y+c*z);
    //return ((x - a * d) * (x - a * d) + (y - b * d) * (y - b * d) + (z - c * d) * (z - c
    * d) <= R * R);
    return ((x - x0) * (x - x0) + (y - y0) * (y - y0) < R * R
    && z<40
    && z>10);
}

```

```

public Boolean isMatched(int i, int j, int k)
{
    //return ((i + j + k) % 2 ==0);
}

```

```

        return ((i % 2 == 1) && (j % 2 == 1) && (k % 2 == 1)) //&& volume[i, j, k]
% 2 == 0);
        || ((i % 2 == 0) && (j % 2 == 0) && (k % 2 == 0)) ;//&& volume[i,
j, k] % 2 == 0);
    }

```

```

public void Input()
{
    Random rand = new Random(1);
    for (int i = 0; i < dimX; i++)
    {
        for (int j = 0; j < dimY; j++)
        {
            for(int k = 0; k < dimZ; k++)
            {

                if (rand.NextDouble() > koevr)
                {
                    volume[i, j, k] = 1;
                }
                else
                {
                    volume[i, j, k] = 0;
                }

            }
        }
    }
}

```

```

}

public void MakeGride()
{
    for (int i = 0; i < dimX; i++)
    {
        for (int j = 0; j < dimY; j++)
        {
            for (int k = 0; k < dimZ; k++)
            {
                if (isMatched(i, j, k) && isRod(i, j, k))
                {
                    volume[i, j, k] = 100;
                    //Console.WriteLine(volume[i,j,k]);
                }
            }
        }
    }
}

public void List()
{
    TextWriter writer = File.CreateText("path.dat");

    for (int i=0;i<dimX;i++)
    {
        for (int j = 0; j < dimY; j++)
        {
            for(int k = 0; k < dimZ; k++)
            {

```



```

        if(volume[i,j,k] !=0)
        {
            writer.WriteLine(String.Format("{0} {1} {2}", i, j, k));
            //Console.WriteLine(i + " " + j + " " + k);
        }
    }
}
Console.WriteLine(i);
}
writer.Flush();
writer.Close();
}

```

```

public string toVMDCoordinate(int i)
{
    String res = "";

    res = i.ToString();
    while(res.Length < 3)
    {
        res = "0" + res;
    }
    res = res + ".000";

    return (res);
}

```

```

public void ListToVMD()
{

```

```

        TextWriter          writer          =
File.CreateText(@"C:\Users\Misha\Desktop\path.pdb");

        for (int i = 0; i < dimX; i++)
        {
            for (int j = 0; j < dimY; j++)
            {
                for (int k = 0; k < dimZ; k++)
                {
                    if (volume[i, j, k] != 0)
                    {
                        writer.WriteLine("ATOM      100  N   VAL A   25      " +
toVMDCoordinate(i) + " " + toVMDCoordinate(j) + " " + toVMDCoordinate(k) + "
1.00 12.00    A1  C   ");
                        //Console.WriteLine(i + " " + j + " " + k);
                    }
                }
            }
            Console.WriteLine(i);
        }
        writer.Flush();
        writer.Close();
    }

```

```

public void tryDraw()
{
    for(int i = 0; i < dimX; i++)

```

```

    {
        for(int j = 0; j < dimY; j++)
        {

            if (volume[i, j, 2] != 0)
            {
                //Console.WriteLine(volume[i,j,25]);
                Console.Write(" X ");
            }
            else
            {
                Console.Write(" 0 ");
            }
        }
        Console.WriteLine("|");

    }

}
}
}
}

```

```
using System;
```

```
namespace Молекулки.GridStuff
```

```

{
    class VPoint

```

```
{  
    public int x, y, z;  
  
    public VPoint()  
    {  
        x = 0;  
        y = 0;  
        z = 0;  
    }  
  
    public VPoint(int X, int Y, int Z)  
    {  
        x = X;  
        y = Y;  
        z = Z;  
    }  
  
    public String ToString()  
    {  
        return String.Format("{0}; {1}; {2}; ", x, y, z);  
    }  
}  
}
```

```
using System;
```

```
using System.IO;
```

```
namespace Молекулки.GridStuff
```

```
{
```

```
    class VolumeCenteredGridHelper
```

```
    {
```

```
        public VolumeCenteredGrid grid;
```

```
        public VolumeCenteredGridHelper()
```

```
        {
```

```
        }
```

```
        public void ExportForVMD(string path)
```

```
        {
```

```
            TextWriter writer = File.CreateText(path);
```

```
            //for (int i = 0; i < grid.dimX; i++)
```

```
            //{
```

```
            //    for (int j = 0; j < grid.dimY; j++)
```

```
            //    {
```

```
            //        for (int k = 0; k < grid.dimZ; k++)
```

```
            //        {
```

```
            //            if (grid.volume[i, j, k])
```

```
            //            {
```

```

//          writer.WriteLine("ATOM    100  N   VAL A   25      " +
toVMDCoordinate(i) + " " + toVMDCoordinate(j) + " " + toVMDCoordinate(k) + "
1.00 12.00    A1  C  ");
//          //Console.WriteLine(i + " " + j + " " + k);
//      }
//  }
// }
// Console.WriteLine(i);
//}

foreach(VPoint p in grid.molecules)
{
    writer.WriteLine("ATOM        100  N   VAL A   25          " +
toVMDCoordinate(p.x)  + "   "  + toVMDCoordinate(p.y)  + "   "  +
toVMDCoordinate(p.z) + " 1.00 12.00    A1  C  ");
}

writer.Flush();
writer.Close();
}

public void ExportForVMDBounded(string path)
{
    TextWriter writer = File.CreateText(path);

    foreach (VPoint p in grid.molecules)
    {

```

```

        if (grid.getNeighbourCount(p.x,p.y,p.z)>0)
        {
            writer.WriteLine("ATOM      100   N      VAL A   25           " +
toVMDCoordinate(p.x) + "   " + toVMDCoordinate(p.y) + "   " +
toVMDCoordinate(p.z) + "  1.00 12.00   A1   C   ");
        }
    }
}

```

```

        writer.Flush();
        writer.Close();
    }

    public string toVMDCoordinate(int coord)
    {
        String res = "";

        res = coord.ToString();
        while (res.Length < 3)
        {
            res = "0" + res;
        }
        res = res + ".000";

        return (res);
    }
}

```

```

    public void printArrayToConsole(int k)
    {
        for(int i = 0; i< grid.dimX; i++)
        {

```

```

for(int j = 0; j < grid.dimY; j++)
{
    if(grid.volume[i, j, k])
    {
        Console.Write("X");
    }
    else
    {
        Console.Write("-");
    }
}
Console.WriteLine();
}
}

```

```

public void MixedArray()
{

```

```

    int x, y;
    VPoint temp;
    for (int i = 0; i < grid.molecules.Count; i++)
    {
        x = grid.random.Next(grid.molecules.Count - 1);
        y = grid.random.Next(grid.molecules.Count - 1);

        temp = grid.molecules[x];
        grid.molecules[x] = grid.molecules[y];
        grid.molecules[y] = temp;
    }
}

```



```

    }
}

```

```

using System;
using System.IO;

```

```

namespace Молекулки.GridStuff
{
    class VolumeCenteredGridHelper
    {
        public VolumeCenteredGrid grid;

        public VolumeCenteredGridHelper()
        {

        }

        public void ExportForVMD(string path)
        {
            TextWriter writer = File.CreateText(path);

            //for (int i = 0; i < grid.dimX; i++)
            //{
            //    for (int j = 0; j < grid.dimY; j++)
            //    {
            //        double x = grid.x[i][j];
            //        double y = grid.y[i][j];
            //        double z = grid.z[i][j];
            //        double v = grid.v[i][j];
            //        writer.WriteLine($"{x} {y} {z} {v}");
            //    }
            //}
        }
    }
}

```

```

//  {
//      for (int k = 0; k < grid.dimZ; k++)
//      {
//          if (grid.volume[i, j, k])
//          {
//              writer.WriteLine("ATOM    100  N    VAL A   25      " +
toVMDCoordinate(i) + " " + toVMDCoordinate(j) + " " + toVMDCoordinate(k) + "
1.00 12.00    A1  C  ");
//              //Console.WriteLine(i + " " + j + " " + k);
//          }
//      }
//  }
//  Console.WriteLine(i);
//}

foreach(VPoint p in grid.molecules)
{
    writer.WriteLine("ATOM        100  N    VAL A   25      " +
toVMDCoordinate(p.x) + " " + toVMDCoordinate(p.y) + " " +
toVMDCoordinate(p.z) + " 1.00 12.00    A1  C  ");
}

writer.Flush();
writer.Close();
}

public void ExportForVMDBounded(string path)
{
    TextWriter writer = File.CreateText(path);

```

```

foreach (VPoint p in grid.moleculas)
{
    if (grid.getNeirbourghCount(p.x,p.y,p.z)>0)
    {
        writer.WriteLine("ATOM      100   N      VAL A   25           " +
toVMDCoordinate(p.x) + "   " + toVMDCoordinate(p.y) + "   " +
toVMDCoordinate(p.z) + " 1.00 12.00    A1   C   ");
    }
}

writer.Flush();
writer.Close();
}

public string toVMDCoordinate(int coord)
{
    String res = "";

    res = coord.ToString();
    while (res.Length < 3)
    {
        res = "0" + res;
    }
    res = res + ".000";

    return (res);
}

```

```
public void printArrayToConsole(int k)
{
    for(int i = 0; i < grid.dimX; i++)
    {
        for(int j = 0; j < grid.dimY; j++)
        {
            if(grid.volume[i, j, k])
            {
                Console.Write("X");
            }
            else
            {
                Console.Write("-");
            }
        }
        Console.WriteLine();
    }
}

public void MixedArray()
{
    int x, y;
    VPoint temp;
    for (int i = 0; i < grid.moleculas.Count; i++)
    {
        x = grid.random.Next(grid.moleculas.Count - 1);
        y = grid.random.Next(grid.moleculas.Count - 1);
    }
}
```

```

        temp = grid.moleculas[x];
        grid.moleculas[x] = grid.moleculas[y];
        grid.moleculas[y] = temp;
    }
}

}
}

```

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading;
using System.Threading.Tasks;
using Молекулки.GridStuff;

```

```

namespace Молекулки

```

```

{
    class Program
    {
        public static string path = @"path.pdb";

        static void Main(string[] args)
        {
            int r = 30;
            VolumeCenteredGrid grid = new VolumeCenteredGrid(200, 200, 300);
            VolumeCenteredGridHelper helper = new VolumeCenteredGridHelper();

```

```

helper.grid = grid;
#region

for (int i = 0; i < grid.dimX; i++)
{
    for (int j = 0; j < grid.dimY; j++)
    {
        for (int k = 0; k < 200; k++)
        {
            r = (int)(16 - 0.0 / 200.0 * k);
            //if ((i - grid.dimX / 2) * (i - grid.dimX / 2) + (j - grid.dimY / 2) * (j -
grid.dimY / 2) + (k - grid.dimZ / 2) * (k - grid.dimZ / 2) <= r * r)
                if ((i - grid.dimX / 2) * (i - grid.dimX / 2) + (j - grid.dimY / 2) * (j -
grid.dimY / 2) <= r * r)
                {
                    if (k < 7)
                    {
                        grid.setPointUnchecked(i, j, k);
                    }
                    else
                    {
                        grid.setPoint(i, j, k);
                    }
                }
            }
        }
    }
}

helper.ExportForVMD(path);

```

```

#endregion
Console.WriteLine(grid.moleculas.Count);

int step = 0;

while (true)
{

    for (int i = 0; i < 1000; i++)
    {
        for (int j = 0; j < grid.moleculas.Count; j++)
        {

helper.grid.jump(helper.grid.moleculas[grid.random.Next(grid.moleculas.Count)]);

        }

        step++;
    }
    Console.WriteLine("step done " + step);

    // helper.ExportForVMD(path);
    helper.ExportForVMDBounded(step+path);
    //export

}
}

static void MainOld(string[] args)
{
    int r = 30;

```

```

VolumeCenteredGrid grid = new VolumeCenteredGrid(200, 200, 200);
#region

for (int i = 0; i < grid.dimX; i++)
{
    for (int j = 0; j < grid.dimY; j++)
    {
        for (int k = 0; k < grid.dimZ; k++)
        {
            if ((i - grid.dimX / 2) * (i - grid.dimX / 2) + (j - grid.dimY / 2) * (j -
grid.dimY / 2) + (k - grid.dimZ / 2) * (k - grid.dimZ / 2) <= r * r)
            {
                grid.setPoint(i, j, k);
            }
        }
    }
}

#endregion

Console.WriteLine(grid.molecules.Count);

VolumeCenteredGridHelper helper = new VolumeCenteredGridHelper();
helper.grid = grid;

int step = 0;

while (true)
{

```



```

    for (int k = 0; k < 1; k++)
    {
        for (int i = 0; i < 1000; i++)
        {
            for (int j = 0; j < grid.moleculas.Count; j++)
            {
                helper.grid.jump(helper.grid.moleculas[j]);
            }

            step++;
        }
        Console.WriteLine("step done " + step);
        helper.MixedArray();
    }

    helper.ExportForVMD(path);

}
Console.ReadKey();
}

public static VolumeCenteredGrid CreateGrid()
{
    int dim = 40;
    VolumeCenteredGrid grid = new VolumeCenteredGrid(5, 5, 2);
    //grid.setPoint(2, 6, 7);

    grid.setPoint(0, 0, 0);

    //for (int i = 0; i < dim; i++)

```

```

//{
//  for (int j = 0; j < dim; j++)
//  {
//    for (int k = 0; k < dim; k++)
//    {
//      grid.setPoint(i, j, k);
//    }
//  }
//}

return grid;
}

public static Boolean InitialConditions(int x, int y, int z)
{
  double a = 1, b = 1, c = 1, d, r = 30, off = 0; //off - відхилення від 0

  d = (a * x + b * y + c * z) / (a * a + b * b + c * c);

  return (true
    // && x < 5 && y < 5 && z < 5

    //&& a*x + b*y + c*z >= 150
    //&& a*x + b*y + c*z <= 250
    //&& (x - off - a * d) * (x - off - a * d) + (y - off - b * d) * (y - off - b * d) +
    (z - off - b * d) * (z - off - b * d) <= r * r
    //|| (x == 0 && y == 0)
    //|| (y == 0 && z == 0)
    //|| (x == 0 && z == 0)
  );
}

```

```
    /// (a * x + b * y + c * z == 250)
    /// (a * x + b * y + c * z == 150)
  );
}

}

}
```